

**2002**

**NASA FACULTY FELLOWSHIP PROGRAM**

**MARSHALL SPACE FLIGHT CENTER  
THE UNIVERSITY OF ALABAMA**

**ADAPTIVE CRITIC NEURAL NETWORK BASED TERMINAL AREA ENERGY  
MANAGEMENT AND APPROACH AND LANDING GUIDANCE**

Prepared By:	Katie Grantham
Academic Rank:	Masters Student
Institution and Department:	University of Missouri-Rolla Department of Aerospace Engineering
NASA/MFSC Directorate:	Space Transportation
MSFC Colleague:	Dr. John Hanson

## **Introduction**

Reusable Launch Vehicles (RLVs) have different mission requirements than the Space Shuttle, which is used for benchmark guidance design. Therefore, alternative Terminal Area Energy Management (TAEM) and Approach and Landing (A/L) Guidance schemes can be examined in the interest of cost reduction. A neural network based solution for a finite horizon trajectory optimization problem is presented in this paper. In this approach the optimal trajectory of the vehicle is produced by adaptive critic based neural networks, which were trained off-line to maintain a gradual glideslope.

## **Problem Formulation and Solution Development**

The point mass equations of motions over a flat Earth are used as the model for the trajectory propagation dynamics. The final time for a TAEM & A/L trajectory varies for each set of initial conditions. Therefore it is more beneficial to use an independent variable whose final value is invariant. Also, the independent variable must be monotonically increasing or decreasing. This led to the reformulation of the vehicle dynamics with altitude as an independent variable. This procedure is shown in Equation 1, where the  $i$ th state equation,  $dX_i/dx$ , where  $X_i$  represents the  $i$ th state has the following form,  $f_i$  is the right hand side of the equations of motion and  $\tau$  represents time. This procedure reduces the order of the vehicle system to five because the state variable  $x$  is represented as the independent variable [5]. The current TAEM and A/L guidance

$$\frac{\partial X_i}{\partial x} = \frac{\partial X_i}{\partial \tau} \frac{\partial \tau}{\partial x} = \frac{f_i}{V \sin \gamma} \quad (1)$$

is divided into flight phases. Those flight phases are Acquisition, Heading Alignment Cone (HAC), Pre-Final, and A/L. The cost function for the trajectory generation, as shown in Equation 2, was selected to minimize the controlled parameters,  $\delta C_L$  and  $\delta \sigma$ , to operate the system near

$$\begin{aligned} J &= \frac{1}{2} \int_{h+H}^h \delta C_L^2 + \delta \sigma^2 dh \\ C_L &= C_{L_{ss}} + \delta C_L \\ \sigma &= \sigma_{ss} + \delta \sigma \end{aligned} \quad (2)$$

steady state,  $\delta C_{L_{ss}}$  and  $\delta \sigma_{ss}$ , during each flight phase. Each flight phase has different goals to accomplish; therefore, an optimal trajectory is one in which each trajectory phase was optimized according to its goal. To accomplish this the selected steady state values of the controls vary for the different trajectory phases. Table 1 shows the various steady state values for each flight phase considered in this formulation [6]. Also, since the flight phases are defined by altitude ranges the cost function transitions are simple to implement [7].

Table 1: Cost Function Parameters				
	Acquisition	HAC	Pre-Final	A/L
$C_{Lss}$	$f(\dot{\gamma} = 0)$			
$\sigma_{ss}$	$f(\psi - \psi_{des})$ $\psi_{des} = \frac{V_{horizontal}}{TurnRadius}$		$f(\psi = 0)$	
	$TR = RF + R_1\psi + R_2\psi^2$	$TR = \frac{V_{horizontal}V}{g \tan \sigma}$		
<b>Terminal Constraints</b>	states <sub>f</sub> =states <sub>HACi</sub>	states <sub>f</sub> =states <sub>p-fi</sub>	$\psi_f=0$ states <sub>f</sub> =states <sub>A/Li</sub>	$\psi_f=0$ $V_f \sin \gamma_f = -2$ $h_f = x_f = y_f = 0$

### Neural Network Solution Development

An action network and a critic network comprise the adaptive-critic neural network structure. The action network behaves as a controller while the critic network behaves as a supervisory network, which criticizes, or evaluates, the outputs of the action network [3]. The system model is used with the cost function to find an optimal control,  $u$ , such that the cost function is minimized. The optimal control problem can be formulated in terms of Hamiltonian [2]. The propagation equations for the Lagrange multipliers, co-states, are subjected to the boundary condition and used along with the system model and the control equations derived from the optimality condition to generate the optimal control for the system. The neural network method of solution development is shown in Figure (1). In this figure the outputs of the ‘‘CRITIC’’

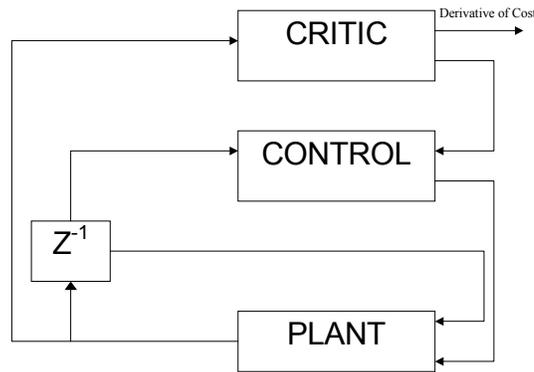


Figure 1: Neural Network Solution Development

block are the co-states the output of the ‘‘CONTROL’’ block is obtained through the optimality condition, and finally the ‘‘PLANT’’ block represents the system model [1]. The solution of finite-horizon problems with neural networks evolves in two stages: the synthesis of the last network and the propagation of the remaining networks [4].

### Synthesis of Last Network

The synthesis of the last network is shown in Figure 2 and described below.

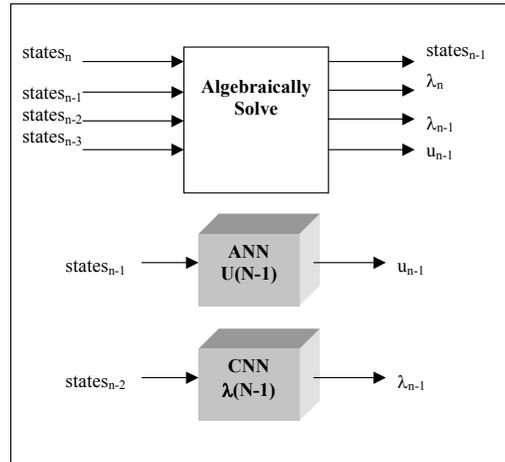


Figure 2: Synthesis of Last Network

1. Generate various random values of  $states_{N-1}$ ,  $states_{N-2}$ ,  $states_{N-3}$  and calculate  $\lambda_N$ ,  $\lambda_{N-1}$ ,  $u_{N-1}$ , the co-states and optimal control respectively, by solving the discrete algebraic propagation and optimality equations.
2. Train two neural networks, the action network (ANN) and the critic network (CNN). The ANN inputs values of  $states_{N-1}$  and outputs  $u_{N-1}$ . The CNN also inputs  $states_{N-2}$  and it outputs  $\lambda_{N-1}$ .

### Synthesis of Remaining Networks

The process of forming the other networks is described below and shown in Figure 3.

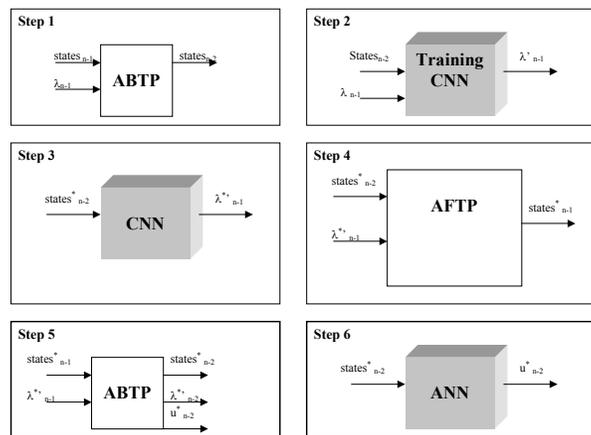


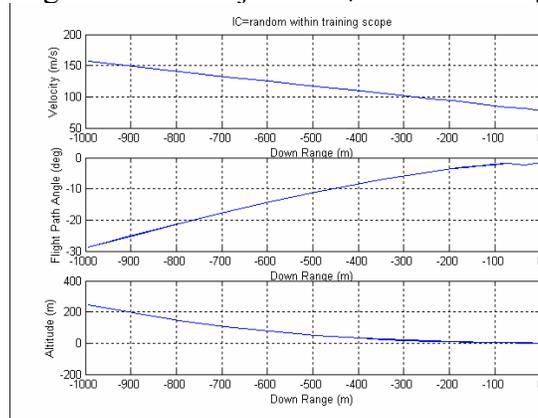
Figure 3: Synthesis of Remaining Networks

1. Algebraically back propagate the trajectory (ABTP) with the  $states_{N-1}$  and  $\lambda_{N-1}$  to get the corresponding  $states_{N-1}$ .

2. Train the CNN with the states<sub>N-2</sub> and the corresponding  $\lambda_{N-1}$ .
3. Expand states<sub>N-2</sub> and use the CNN to obtain the corresponding  $\lambda_{N-1}$ .
4. Algebraically forward propagate the trajectory (AFTP) with the expanded states<sub>N-2</sub> and the corresponding  $\lambda_{N-1}$  to obtain the corresponding states<sub>N-1</sub>.
5. ABTP with the new states<sub>N-1</sub> to obtain states<sub>N-2</sub>,  $\lambda_{N-2}$ , and  $u_{N-2}$ .
6. Train an ANN with the states<sub>N-2</sub> as input and  $u_{N-2}$  as output.

### Results

The neural network synthesis as described above was used to generate TAEM and A/L trajectories. The trajectories generated used down range as an independent variable, rather than altitude, as discussed earlier so that the results can first be validated with an existing result. The performance of the neural network generated trajectories is obtained by assuming any state variable value, within the trained scope, and using the appropriate  $\delta C_L$  lift coefficient correction factor during simulations to generate the trajectories. The down range history from  $-1000$  m of

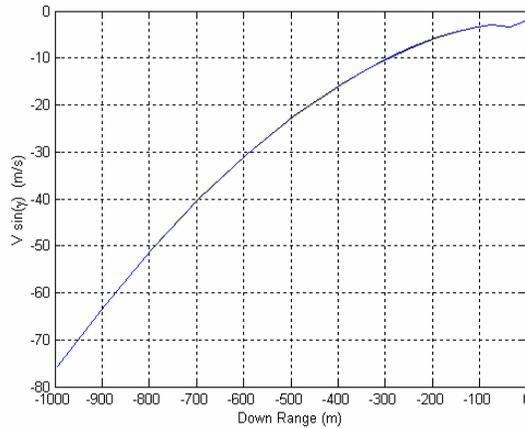


**Figure 4: Down Range Neural Network Trajectory Histories**

the velocity, the flight path angle, and the altitude for various initial conditions are shown in Figure 4. The ranges of these initial conditions are provided in Table 2. These results were

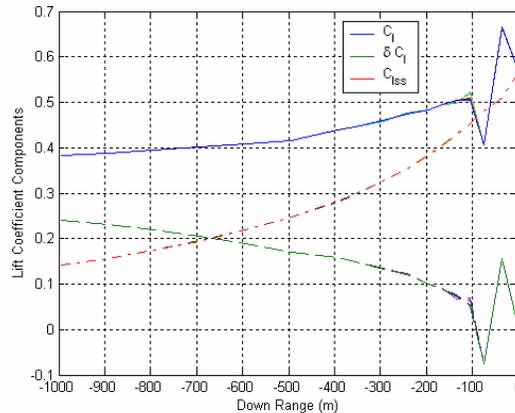
Table 2: Initial Condition Ranges (from $-1000$ m Down Range)		
	Maximum	Minimum
Velocity (m/s)	157	156
Flight Path Angle (deg)	-28.8	-29.7
Altitude (m)	248	241

generated without the use of the CNN as an expander of initial conditions. This was done so that this trajectory set could be compared to an existing trajectory to determine its validity. Figure 5 demonstrates that from  $-1000$  meters before the runway the RLV can meet the final constraints,  $x_f=0$ , and  $V_f \sin \gamma_f = -2$  m/s for any initial condition within the predefined range. The down range history of lift coefficient,  $C_L$ , which is used as the guidance control variable is displayed in.



**Figure 5: Satisfaction of Final Constraints**

Figure 6 along with the down range history of the lift coefficient components. From this figure it is determined that the commanded lift coefficient values are within an acceptable range for a RLV



**Figure 6: Control History**

**Future Work**

Once the trajectory comparison has been completed, the CNN expansion, as described in Section 3.2 will be applied. After its successful application, the study will transition into the use of altitude as the independent variable and the 6-degrees of freedom trajectory synthesis will commence. This study will continue until the TAEM interface is reached with the neural network synthesis. The final results of this study will produce a larger range of initial conditions, which like trajectories shown in Figure 5, will meet the predetermined final constraints. Finally the neural network guidance will be implemented into Marshall Aerospace Vehicle Representation In C (MAVERIC) for a final validation on the X-33 aircraft simulation.

**Conclusion**

A neural network approach to developing optimal TAEM trajectories has been presented in this study. This approach solves a nonlinear guidance problem without linearizing the model. The results of the future work will demonstrate the powerful capability of adaptive critic based neural networks to solve this class of problems.

### **Acknowledgements**

The author would like to thank the NASA Faculty Fellowship program and John Hanson for the opportunity to do research at the Marshall Space Flight Center. Also, Warren Adams, Greg Dukeman, and Jim McCarter for their help with MAVERIC. As well as S.N. Balakrishnan for his wonderful performance as an academic advisor.

### **References**

[1] Balakrishnan, S. N. and Biega, V., “Adaptive Critic Based Neural Networks for Aircraft Optimal Control”, *Journal of Guidance, Control, and Dynamics*. Vol. 19, No .4, July-Aug, 1999, pp893-898.

[2] Bryson, A. E. and Ho, Y. C., *Applied Optimal Control*, Halsted Press, 1975.

[3] Hagan, M.T., Demuth, H.B. and Beale, M., “Neural Network Design”. PWS Publishing Company, Boston, 1996,(pp1-1).

[4] Han, Dongchen and Balakrishnan,S.N., “Adaptive Critic-Based Neural Networks for Agile Missile Control”, *Journal of Guidance, Control, and Dynamics*, Vol. 25, No.2, March-April, 2002,pp404-407.

[5] Lu, Ping, Shen, Zuojun, Dukeman, Greg A., Hanson, John, M., “Entry Guidance by Trajectory Regulation”, *Proceedings of AIAA Guidance, Navigation, and Control Conference*, Denver, CO, August 14-17, 2000.

[6] Moore, Thomas, “Space Shuttle Entry Terminal Area Energy Management”, NASA Technical Memorandum 104744, November 1991.

[7] Olson, Lee, “Flight Systems Design and Analysis, Terminal Area Energy Management and Approach/Landing Guidance and Control”, Presentation, X-33, Day 3, FA, July 1996, pp250-276.